Dimensionality reduction in product of metric spaces

Aleksander Denisiuk[0000−0002−7501−7048]

University of Warmia and Mazury in Olsztyn
ul. Słoneczna 54, 10-710 Olsztyn, Poland
denisiuk@matman.uwm.edu.pl

Abstract. The purpose of the article is to develop a new dimensionality reduction algorithm for data that are described by many features of different nature. A method of feature selection is based on a new concept of metrical importance of the features. The concept of feature importance is based on metrical properties of data and is inspired by the principle component analysis. Numerical experiments confirm the effectiveness of the method and certain accordance of it with other concepts of feature importance.

Keywords: dimensionality reduction · feature selection · feature importance · explainable machine learning · metric learning · weighted metric · classification.

1 Introduction

Dimensionality reduction of the data space while retaining as much information as possible is important problem in the data analysis. Beside other things, it reduces computational complexity of various algorithms, mitigates the curse of dimensionality, and thus has many applications in clustering, classification, visualization, and compression of high-dimensional data (see, for instance, the survey [20]).

The purpose of this article is further development of the dimensionality reduction method from [8] proposed for categorical data. That method was inspired by the classical linear PCA feature selection. Namely, it was shown in [8] that PCA has the following metrical interpretation. Consider the affine transform that minimizes the total squared inner-class distance. It turns out that the major feature is scaled with the minimal multiplier, the minor feature—with the maximal one, like at figure 1. This interpretation was transferred to the categorical data space with the Hamming metic, and problem of feature selection was reduced to certain linear programming problem. See [8] for more details.

In this article the same interpretation of the PCA leads to formulation of appropriate non-linear optimization problem in product of metrical spaces. The solution of the optimization problem allows to calculate multipliers that are interpreted as the feature importances. In what follows it called the *metrical*

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97629-2_20

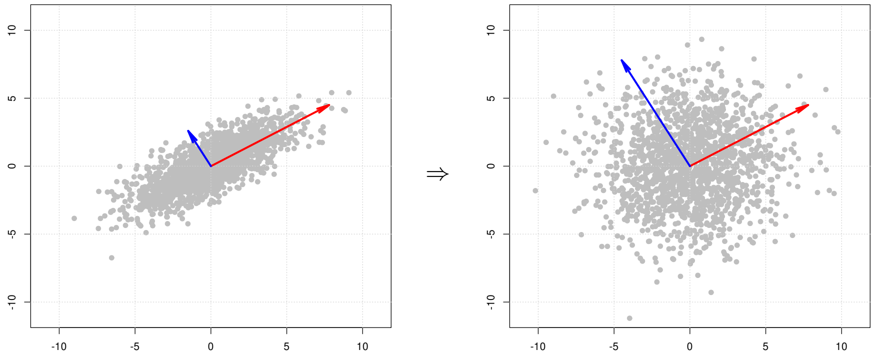

Fig. 1. Scaling that minimizes total relative inner-class squared distance

importances. In such a way, dimensionality reduction consists of dropping less important features first.

To prove the concept, numeric experiments on three datasets are performed. The data features were discarded in order of growing importances and the F_1Score of classification was measured. Beside new introduced metrical importances other known feature importances were also considered: random forest mean decrease in accuracy, mean decrease in Gini index, the “standard errors” of the permutation-based importance measure, and the Shapley values. In all cases the average value of feature importance with respect to all classes was considered.

Experiments show that despite the importance values of individual features for different methods are different, all the methods show similar efficiency. It should also be mentioned that the metrical importance is much simpler to calculate. In fact, the metrical importance is calculated with two explicit formulas: (1) and (5). The computational time for the Shapley values increases exponentially with the number of features [15, Chapter 17]. Calculation of random forest related features importance involves building of decision trees and grows significally with number of features [5].

The rest of the paper is organized as follows. In section 2 there is a short survey of the basic related works. Section 3 contains the proposed algorithm for metrical importances calculation. The performed numerical experiments are discussed in the section 4. Finally, some concluding remarks are given in the section 5.

2 Related Works

The key notion of new algorithm is the metrical importance of a feature. Importances of data features are of active study in recent years in the framework of explainable machine learning. One can find a comprehensive review in recently published book [15]. Here only some concepts will be mentioned. Three importances related to the random forest classifier: mean decrease in accuracy, the mean decrease in Gini index, and the "standard errors" of the permutation-based importance measure [5]. The last one concept is based on the Shapley values that were introduced in [19] for the game theory. In the article [21] the authors suggest to interpret the Shapley values as a contribution of individual feature to data classification. Metric-based importance comes from the principle components analysis and is direct continuation of the work [8].

All the mentioned concepts of feature importance are used in numerical experiments in section 4.

Another field of machine learning that concerns this work is the metric learning. The current state of the metric learning can be found in surveys [4] and [12]. Most of methods concern data with pure numerical features. Non-numerical features are often embedded into continuous space. Some papers develop methods of metric learning for structured data: graphs with the graph-editing metric or text strings with the Levenshtein distance [17, 3]. The proposed in this article approach can be used to data with mixed features of any nature and metric.

Determining the weights assigned to individual features of mixed data was recently used in context of supervised and unsupervised machine learning respectively in articles [7] and [9].

3 Metrical importance of the features and algorithm for dimensionality reduction

Assume that the dataset $\mathbf{X}$ of M instances is given. Let each instance $x \in \mathbf{X}$ has n features of different kind, $x = (x_1, \ldots, x_n)$. Suppose that each feature x_i, $i = 1, \ldots, n$ is equipped with an appropriate distance $\mathrm{dist}_i(\cdot, \cdot)$ that measures dissimilarity of the data.

The product distance on $\mathbf{X}$ in defined in the following way:

$$\mathrm{dist}^2(x, y) = \sum_{i=1}^{n} \mathrm{dist}_i^2(x_i, y_i).$$

Introduce the weights vector $u = (u_1, \ldots, u_n) \in \mathbb{R}^n$, where $u_i \geq 0$ corresponds to a *scaling* of the feature i for $i = 1, \ldots, n$. The weighted distance is defined as follows:

$$\mathrm{dist}_u^2(x, y) = \sum_{i=1}^{n} u_i^2 \,\mathrm{dist}_i^2(x_i, y_i).$$

The last assumption is that the dataset is divided into c classes, $\mathbf{X} = C_1 \cup \cdots \cup C_c$.

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97629-2_20

The total inner-class squared distance is

$$G(u) = \frac{1}{M^2} \sum_{k=1}^{c} \sum_{x,y\in C_k} \operatorname{dist}_u^2(x,y) = \frac{1}{M^2} \sum_{k=1}^{c} \sum_{x,y\in C_k} \sum_{i=1}^{n} u_i^2 \operatorname{dist}_i^2(x_i,y_i)$$
$$= \sum_{i=1}^{n} u_i^2 \left(\frac{1}{M^2} \sum_{k=1}^{c} \sum_{x,y\in C_k} \operatorname{dist}_i^2(x_i,y_i) \right) = \sum_{i=1}^{n} u_i^2 z_i,$$

where

$$z_i = \frac{1}{M^2} \sum_{k=1}^{c} \sum_{x,y\in C_k} \operatorname{dist}_i^2(x_i,y_i). \tag{1}$$

Consider the following constraint minimization problem:

$$\begin{cases} G(u) = \sum\limits_{i=1}^{n} u_i^2 z_i \to \min, \\ \frac{1}{n} \sum\limits_{i=1}^{n} u_i = 1, \quad u_i \geq 0 \text{ for } i = 1,\ldots,n. \end{cases} \tag{2}$$

Definition 1. *The* metrical importance I_m *of the feature* $i = 1,\ldots,n$ *equals*

$$I_m(i) = 1/u_i^0,$$

where $u^0 = (u_1^0,\ldots,u_n^0)$ *is the solution of the minimization problem (2).*

Remark 1. Consider multivariate normally distributed data with distribution function

$$f(x) = \frac{\exp\left(-\frac{1}{2}x^T \Sigma^{-1} x\right)}{\sqrt{(2\pi)^n \det \Sigma}},$$

where Σ is diagonal matrix, $\Sigma = \operatorname{diag}(\lambda_1,\ldots,\lambda_n)$, $\Sigma^{-1} = \operatorname{diag}(\lambda_1^{-1},\ldots,\lambda_n^{-1})$, $\det\Sigma = \lambda_1\cdots\lambda_n$, and $\lambda_i > 0$ for $i = 1,\ldots,n$. One can show [8] that solution of the problem (2) is proportional to vector $\lambda = (\lambda_1^{-1},\ldots,\lambda_n^{-1})$. In such a way features with greater variance have greater metrical importance. This is in accordance with the principle components analysis (see, for instance [1]).

Remark 2 (Data normalization). Since different features may change at different ranges, as is the case of most metric-based algorithms, data normalization must be performed. Numerical features can be rescaled such that the mean value will be 0, and variance 1. In this case the mean squared distance $\operatorname{dist}^2(x,y) = (x-y)^2$ is equal to 2. So, for non-numerical features the corresponding distance should be scaled with multiplier to make the mean squared distance also be equal to 2. Specifically, for categorical feature of cardinality k the standard Hamming distance

$$\operatorname{dist}_h(x,y) = \operatorname{diff}(x,y) = \begin{cases} 1, & x \neq y, \\ 0, & x = y, \end{cases}$$

should be multiplied by $\sqrt{2k/(k-1)}$.

ICCS Camera Ready Version 2025
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-97629-2_20

To solve the problem (2) one can use the Lagrange multipliers method. Corresponding Lagrange function is

$$L(u, \Lambda) = \sum_{i=1}^{n} u_i z_i - \Lambda \left(\sum_{i=1}^{n} u_i - n \right). \tag{3}$$

Differentiating (3) with respect to u_i and equalizing the result to zero, one obtains

$$u_i = \frac{\Lambda}{2 z_i}, \quad i = 1, \ldots, n. \tag{4}$$

Differentiation with respect to Λ implies

$$\Lambda = \frac{2n}{\sum_{i=1}^{n} z_i^{-1}}.$$

Substituting this to (4) one gets

$$u_i = \frac{1}{z_i} \left(\frac{1}{n} \sum_{i=1}^{n} z_i^{-1} \right)^{-1}, \quad i = 1, \ldots, n.$$

Finally, for metrical importance (definition 1) one have

$$I_m(i) = z_i \left(\frac{1}{n} \sum_{i=1}^{n} z_i^{-1} \right), \quad i = 1, \ldots, n. \tag{5}$$

The above considerations are summarized in the algorithm 3.1.

Algorithm 3.1 Reduction of m dimensions

```
Require: the dimension of dataset $\mathbf{X}$ is $n$, $n > m$
Ensure: the dimension of dataset $\mathbf{X}$ is $n - m$
  compute coefficients $z_i$ with the formula (1)
  compute importances with the formula (5)
  $s \leftarrow 0$
  while $s < m$ do
    discard the feature with the less importance
    $s \leftarrow s + 1$
  end while
```

4 Numerical Experiments

To illustrate the concept a few R scripts have been created. The code is available as a project on Gitlab at `https://gitlab.com/adenisiuk/l2`.

The purpose of the tests is to show that the introduces in this article algorithm 3.1 allows to reduce dimensionality while retaining as much information as possible. To do this, consider the classification problem. The data first were classified with complete set of features ordered by decreasing importance. Then less important features one by one were discarded and F_1Score of classification was measured. This order is called the *pca* order (red line in the figures).

The method was also considered with other importances mentioned in the section 2. The order that corresponds to the Shapley values is referred as the *shapley* order, the green line on the pictures. The order that corresponds to the mean decrease of accuracy in random forest classifier is plotted in blue color and is referred as the *rf* order. The order related to the decrease in Gini index is referred as the *gini* order (magenta line). And the order generated by the "standard errors" of the permutation-based importance measure for random forest classifier has cyan color and is called as the *sd* order.

The Shapley values were calculated with the `fastshar` R package [10]. The random forest classifier was used as the user-specified prediction wrapper.

Three importances related to the random forest classifier were calculated with the R implementation [13].

The reverse orders for every importance were also tested. They are referred correspondingly as *acp*, *yelpahs*, *fr*, *inig*, *ds*. Related lines in the figures has the same color, but the dashed style.

Implementations of three classifiers: random forest, SVM and XGBoost in R were used in experiments: [13, 14, 6]. For the random forest classifier an average result for 100 tests is presented.

In all the above classifiers the standard implementation settings were used.

3 datasets from the UCI Machine Learning Repository [2] were considered: the Australian Credit Approval [18], the Bank Marketing [16] and the Heart Disease [11]. These datasets contain numerical and categorical data. The standard difference metric for numerical and categorical parts, normalized according to remark 2 was used in experiments.

Note that dataset features were rearranged for the implementation: continuous features are the first, and categorical features are placed at the end of the feature list.

All the tested datasets were split into train (80%) and test (20%) parts.

The F_1Score measure with respect to the first (having a smaller amount of records) class was used to estimate the classification rate.

The considered datasets have only two classes, but the method can be used to dataset with greater number of classes as well.

For each dataset the importances with respect to five considered concepts, corresponding orders of the features discarding and performance of classification after discarding the features were calculated. The importances were calculated with full datasets, and then were scaled to sum to 100%.

One can see that despite the importances and orders are different, in all the tests direct orders have similar performance and give the minimal information loss, while reverse orders give greater lost.

DOI: 10.1007/978-3-031-97629-2_20

4.1 Australian Credit Approval

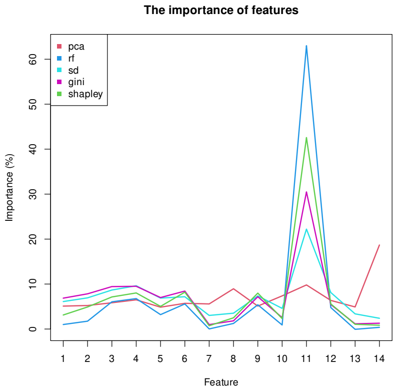

Fig. 2. Importances of features for the Australian Credit Approval dataset

Table 1. Orders of feature discarding for the Australian Credit Approval dataset

	Feature discarding order													
pca	5	13	1	9	2	7	6	3	12	4	10	8	11	14
rf	13	7	14	10	1	8	2	5	12	9	6	3	4	11
sd	14	7	13	8	10	1	5	2	6	9	12	3	4	11
gini	7	13	14	8	10	12	1	5	9	2	6	3	4	11
shapley	7	14	13	10	8	1	2	5	12	3	9	4	6	11

The Australian Credit Approval [18] dataset has 6 continuous, 8 nominal attributes, 690 records, and 2 decision categories.

The importances of individual features and the orders of feature discarding are presented in the figure 2 and the table 1. The F_1Score for selected classifiers and different orders of the feature discarding are presented in the figure 3.

One can see that all the algorithms marked the feature 11 as very important.

Large difference can be observed in the most important feature. The metric based algorithm marked 14 as as most important, while the other algorithms marked 11th feature. The reason probably is that these features are strongly dependent. χ^2 dependency test produced p-value of 0.0006133. Dependency in some sens means that these features contain similar information. So, it should have similar importance. This guess is confirmed if one observe the *acp* order: after discarding the feature 14 and keeping 11, information lost is low.

Other difference one can see analysing the less important feature. The new algorithm mark the feature 5, according to the *rf* order it is the feature 13,

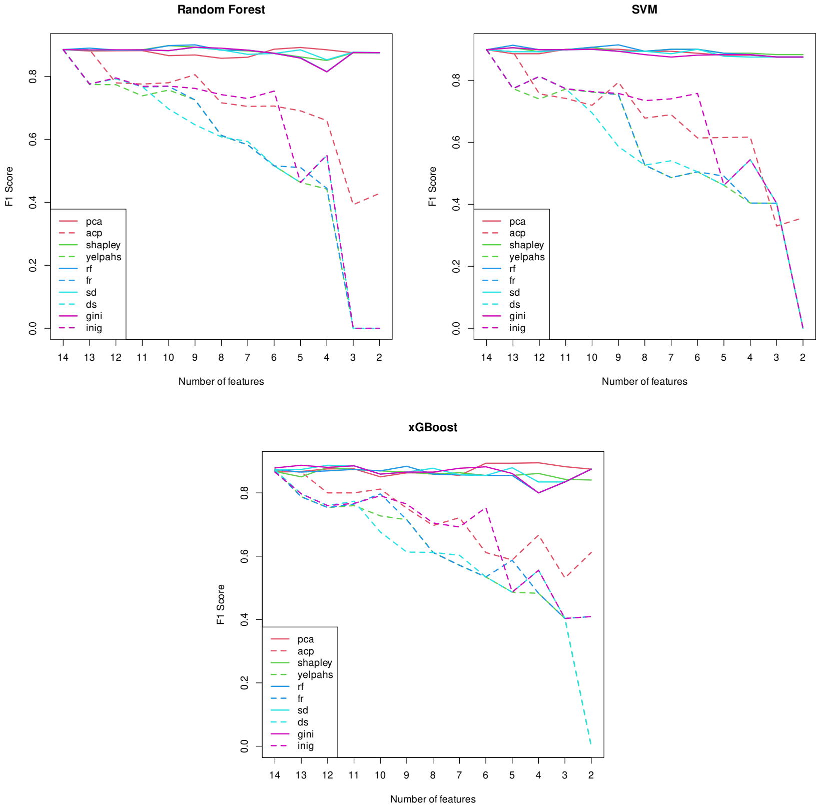

Fig. 3. Classification accuracy for the Australian Credit Approval dataset

sd marked the feature 14, rest of algorithms suggest the feature 7. Despite of all the mentioned differences, the performance of classification after the less important feature discarding is almost identical. That can be corollary of the fact that the levels of importance of less significant features are generally low and approximately the same.

4.2 Bank Marketing

The Bank Marketing dataset [16] has 7 continuous, 9 nominal attributes, 4521 records, 2 decision categories.

The importances of individual features and the orders of feature discarding are presented in the figure 4 and the table 2. The F_1Score for selected classifiers and different orders of the feature discarding are presented in the figure 5.

Again one can observe a difference in the features order, but similar performance for all direct orders. For the random forest classifier the metrical importance order even overperforms the others. Indeed, most of algorithms marked

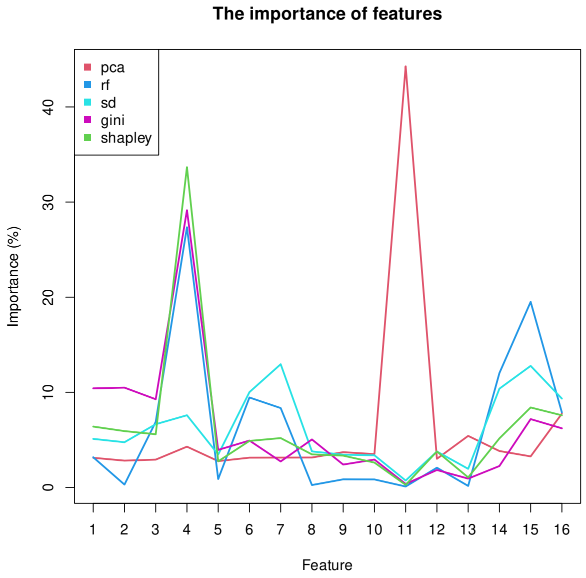

Fig. 4. Importances of features discarding for the Bank Marketing dataset

Table 2. Orders of feature discarding for the Bank Marketing dataset

	Feature discarding order															
pca	5	2	3	12	1	6	7	8	15	10	9	14	4	13	16	11
rf	11	13	8	2	10	9	5	12	1	3	16	7	6	14	15	4
sd	11	13	10	9	5	12	8	2	1	3	4	16	6	14	15	7
gini	11	13	12	14	9	7	10	5	6	8	16	15	3	1	2	4
shapley	11	13	10	5	9	8	12	6	14	7	3	2	1	16	15	4

the feature 11 as less important. But dropping this feature in random forest classifier experiments caused significant lost of F_1Score.

Note, however, that generally performance of classification for this dataset is poor.

4.3 Heart Disease

The Heart Disease dataset [11] has 5 continuous and 8 nominal attributes, 270 records, 2 decision categories.

The importances of individual features and the orders of feature discarding are presented in the figure 6 and the table 3. The F_1Score for selected classifiers and different orders of the feature discarding are presented in the figure 7.

For this dataset one can observe, like in case of the Australian Credit Approval dataset, that the 6th feature was marked as very important for all the

DOI: 10.1007/978-3-031-97629-2_20

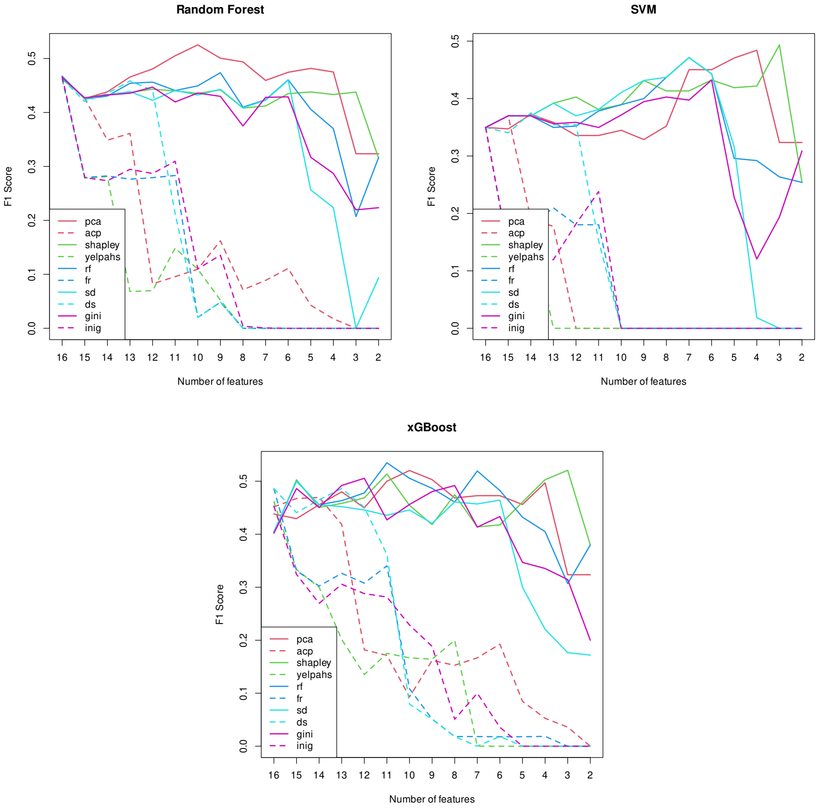

Fig. 5. Classification accuracy for the Bank Marketing dataset

algorithms. But the *pca* marked 8th feature as the most important. And, again, observing reverse order tests one can see that dropping the 6th feature at the first step does not cause the lost of F_1Score, but even results in F_1Score gain.

The following difference between the metrical importance and other concepts should be mentioned. Namely, the metrical importance stems from the metrical properties of the data and is independent of any classifier. The remaining concepts are closely related to the random forest classifier. This difference has two consequences. On the one hand, the *pca* ordering produces more stable results across different classifiers (this can be seen especially in the Australian Credit Approval dataset). On the other hand, if the metric used to compare data does not reflect its structure, the classification results may be unsatisfactory.

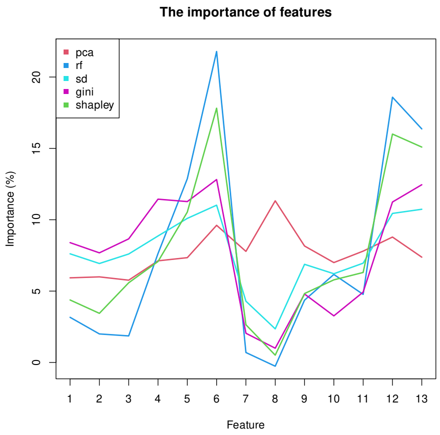

Fig. 6. Importances of features for the Heart Disease dataset

Table 3. Orders of feature discarding for the Heart Disease dataset

	Feature discarding order												
pca	3	1	2	10	4	5	13	7	11	9	12	6	8
rf	8	7	3	2	1	9	11	10	4	5	13	12	6
sd	8	7	10	9	2	11	3	1	4	5	12	13	6
gini	8	7	10	9	11	2	1	3	12	5	4	13	6
shapley	8	7	2	1	9	3	10	11	4	5	13	12	6

5 Conclusion and Future Work

In this article a new concept of metrical importance of data features is proposed. The notion is inspired by the classical PCA, but can be applied to any mixed data with appropriate metric defined on individuals features.

A simple algorithm of dimensionality reduction based on the new notion was developed: discard less important features first.

Numerical experiments with three classifiers were performed: random forest, SVM, XGBoost. Other concepts of feature importance were also considered: three related to the random forest classifier and the Shapley values (see the section 2 for more details.

Experiments show that for considered datasets the new metrical importance produces dimensionality reduction algorithm of efficiency that is close to the above-mentioned knows concepts of feature importance, while the metrical importance has much smaller computational complexity.

So, one can suggest to use the metrical importance as a new instrument in analysis of mixed data.

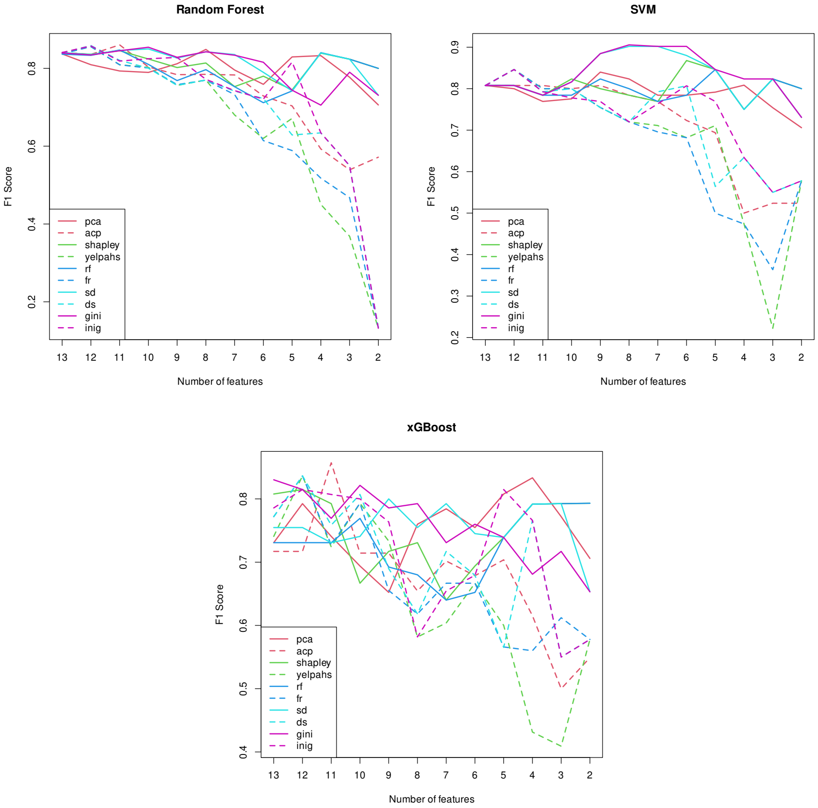

Fig. 7. Classification accuracy for the Heart Disease dataset

The concept of metric importance deserves further development in the context of explainable machine learning. It would be interested to analyse it with real data set.

References

1. Afifi, A., May, S., Donatello, R., Clark, V.: Practical Multivariate Analysis. Chapman & Hall/CRC texts in statistical science series, CRC Press, Taylor & Francis Group (2019), `https://books.google.pl/books?id=AUyrswEACAAJ`
2. Asuncion, A., Newman, D.J.: UCI machine learning repository (2007), `http://www.ics.uci.edu/~mlearn/MLRepository.html`
3. Bellet, A., Habrard, A., Sebban, M.: Good edit similarity learning by loss minimization. Machine Learning **89**, 5–35 (2012). https://doi.org/10.1007/s10994-012-5293-8
4. Bellet, A., Habrard, A., Sebban, M.: Metric learning. Springer Cham (2015). https://doi.org/10.1007/978-3-031-01572-4

5. Breiman, L.: Random forests. Machine learning **45**, 5–32 (2001)
6. Chen, T., He, T., Benesty, M., Khotilovich, V., Tang, Y., Cho, H., Chen, K., Mitchell, R., Cano, I., Zhou, T., Li, M., Xie, J., Lin, M., Geng, Y., Li, Y., Yuan, J.: XGBoost: Extreme Gradient Boosting (2023), `https://CRAN.R-project.org/package=xgboost`, r package version 1.7.6.1
7. Denisiuk, A.: Weighted hamming metric and knn classification of nominal-continuous data. In: Mikyška, J., de Mulatier, C., Paszynski, M., Krzhizhanovskaya, V.V., Dongarra, J.J., Sloot, P.M. (eds.) Computational Science – ICCS 2023. pp. 306–313. Springer Nature Switzerland, Cham (2023)
8. Denisiuk, A.: PCA dimensionality reduction for categorical data. In: Franco, L., de Mulatier, C., Paszynski, M., Krzhizhanovskaya, V.V., Dongarra, J.J., Sloot, P.M.A. (eds.) Computational Science – ICCS 2024. pp. 179–186. Springer Nature Switzerland, Cham (2024)
9. Denisiuk, A., Grabowski, M.: Embedding of the hamming space into a sphere with weighted quadrance metric and c-means clustering of nominal-continuous data. Intelligent Data Analysis **22**(6), 1297001314 (2018). https://doi.org/10.3233/IDA-173645
10. Greenwell, B.: fastshap: Fast Approximate Shapley Values (2024), `https://github.com/bgreenwell/fastshap`, r package version 0.1.1, https://bgreenwell.github.io/fastshap/
11. Janosi, A., Steinbrunn, W., Pfisterer, M., Detrano, R.: Heart Disease. UCI Machine Learning Repository (1989), DOI: https://doi.org/10.24432/C52P4X
12. Kulis, B.: Metric learning: A survey. Foundations and Trends® in Machine Learning **5**(4), 287–364 (2013). https://doi.org/10.1561/2200000019
13. Liaw, A., Wiener, M.: Classification and regression by randomforest. R News **2**(3), 18–22 (2002)
14. Meyer, D., Dimitriadou, E., Hornik, K., Weingessel, A., Leisch, F.: e1071: Misc Functions of the Department of Statistics, Probability Theory Group (Formerly: E1071), TU Wien (2022), r package version 1.7-12
15. Molnar, C.: Interpretable Machine Learning. 2 edn. (2022), `https://christophm.github.io/interpretable-ml-book`
16. Moro, S., Rita, P., Cortez, P.: Bank Marketing. UCI Machine Learning Repository (2014), DOI: https://doi.org/10.24432/C5K306
17. Neuhaus, M., Bunke, H.: Automatic learning of cost functions for graph edit distance. Information Sciences **177**(1), 239–247 (2007). https://doi.org/10.1016/j.ins.2006.02.013
18. Quinlan, R.: Statlog (Australian Credit Approval). UCI Machine Learning Repository (1987), DOI: https://doi.org/10.24432/C59012
19. Shapley, L.S., et al.: A value for n-person games (1953)
20. Van Der Maaten, L., Postma, E., Van den Herik, J.: Dimensionality reduction: a comparative review. Journal of Machine Learning Research **10**, 66–71 (2009)
21. Štrumbelj, E., Kononenko, I.: Explaining prediction models and individual predictions with feature contributions. Knowl. Inf. Syst. **41**(3), 647–665 (Dec 2014). https://doi.org/10.1007/s10115-013-0679-x